the transition crosses the logic threshold. The energy dissipated in the wiring resistance R is

$$E = \int i^2(t) R \, dt \ . \tag{5.2}$$

The current flow during one transition

$$i(t) = \frac{V}{R} \exp\left(-\frac{t}{RC}\right) \ , \tag{5.3}$$

so the dissipated energy per transition (either positive or negative)

$$E = \frac{V^2}{R} \int_0^\infty \exp\left(-\frac{2t}{RC}\right) dt = \frac{1}{2} CV^2 \ . \tag{5.4}$$

If pulses occur at a frequency f, the power dissipated in both transitions

$$P = fCV^2 \ . \tag{5.5}$$

Thus, the power dissipation increases with clock frequency and the square of the logic swing.

Fast logic is time-critical. It relies on logic operations from multiple paths coming together at the right time. Valid results depend on maintaining minimum allowable overlaps (*e.g.* AND) and setup times (latches). Each logic circuit has a finite propagation delay, which depends on circuit loading, *i.e.* how many loads the circuit has to drive. In addition, as illustrated in Figure 5.6 the wiring resistance and capacitive loads introduces delay. This depends on the number of circuits connected to a wire or trace, the length of the trace and the dielectric constant of the substrate material. Relying on control of circuit and wiring delays to maintain timing requires great care, as it depends on circuit variations and temperature. In principle all of this can be simulated, but in complex systems there are too many combinations to test every one. A more robust solution is to use synchronous systems, where the timing of all transitions is determined by a master clock. Generally, this does not provide the utmost speed and requires some additional circuitry, but increases reliability. Nevertheless, clever designers frequently utilize asynchronous logic. Sometimes it succeeds . . . and sometimes it doesn't.

5.1.3 *Logic arrays*

Commodity integrated circuits with basic logic blocks are readily available, *e.g.* with four NAND gates or two flip-flops in one package. These can be combined to form simple digital systems. However, complex logic systems are no longer designed using individual gates. Instead, logic functions are described in a high-level language (*e.g.* VHDL), synthesized using design libraries, and implemented as custom ICs – "ASICs" (application specific ICs) – or programmable logic arrays. In these implementations the digital circuitry no longer appears as an